CLAIMS

1. A data compressor for compressing a data file, comprising:

a set of dictionary lists that comprises a set of statically encoded words;

a dynamic word encoder that generates a set of dynamically encoded words from words in the data file that are not in the set of statically coded words; and

a data encoder that compresses the data file by determining whether words in the data file are in the set of statically encoded words or in the set of dynamically encoded words.

2. The data compressor according to claim 1, wherein the set of dictionary lists comprises a primary dictionary list comprising both a statically encoded word portion and a dynamically encoded word portion.

3. The data compressor according to claim 2, wherein the words in the primary dictionary list are identified by eight bit tokens.

4. The data compressor according to claim 2, wherein the set of dictionary lists comprises a common word dictionary list of at least 50,000 statically encoded words greater than or equal to 4 characters in length.

5. The data compressor according to claim 1, wherein the set of dictionary lists comprises a common word dictionary list of statically encoded common words comprising at least 50,000 words.

6. The data compressor according to claim 5, wherein the common word dictionary list comprises a predetermined number of the most frequent words in a set of test data files.

7. The text compressor according to claim 1, wherein the set of dictionary lists includes a common word dictionary list of statically encoded common words comprising words greater than M characters in length.

8. The text compressor according to claim 5, wherein M is 3.

9. A method used for compressing a data file, comprising in a first computer the steps of:

determining a set of statically coded words to be used in a data compressor; and

storing the set of statically coded words in the data compressor; and

in the data compressor, the steps of:

determining a set of dynamically coded words from words in the data file that are not in the set of statically coded words; and

storing the set of dynamically coded words in a dictionary list of the data compressor.

10. The method according to claim 9, further comprising in the data compressor the steps of:

compressing the data file by determining whether a word in the data file is in the set of statically encoded words or in the set of dynamically encoded words, and when the word is determined to be within one of the sets of encoded words, substituting for the word a token that identifies the word within the sets of encoded words.

11. A method used for constructing a dictionary list used in a data compressor, comprising

in a first computer the steps of:

determining a complete set of statically encoded words to be used in the data compressor; and

storing the complete set of statically encoded words in the data compressor; and

in the test compressor,

determining a set of dynamically encoded words from words in a data file that are not in the complete set of statically encoded words; and

storing the set of dynamically encoded words in a dictionary list of the data compressor.

12. A dictionary word list used in a data compressor, comprising:

a set of words, wherein the set of words is sorted into partitions, and wherein each partition having a common starting character is sorted by lengths of the words, and wherein each partition is identified by the common starting character and the one of the lengths; and

an index, wherein a starting location for each partition is stored.

13. A method used for constructing a dictionary word list used in a data compressor, comprising in a first computer the steps of:

determining a set of words that are included in the dictionary word list;

sorting each subset of words having a common starting character by lengths of the words, wherein each subset of the set of words that was sorted by one of the lengths is named a partition and is identified by the common starting character and the one of the lengths; and

storing each partition at a starting location in memory, and

storing an index of the starting locations for each common starting character and the ones of the lengths.

14. The method according to claim 13, wherein in the step of sorting each set of words having a common starting character by lengths of the words, the lengths of the words start at a minimum length and end at a maximum length, and include all integral lengths from the minimum to the maximum length, and wherein all words that are as long as, or longer than, the maximum length are in one partition.

15. A method of compressing a data file using tokens that identify symbol strings stored in one or more dictionaries, comprising the steps of:

determining that two successive symbol strings in a data file are represented by two tokens that are not necessarily different;

determining that the two successive symbol strings are separated only by a space; and

encoding the two successive symbol strings in a compressed data file by arranging the two tokens adjacently, without an intervening space symbol.